## BHAVANA CK 1BM20CS403 CSE-4A

## Program 9: Movie database

Consider the schema for Movie Database:

ACTOR (Act\_id, Act\_Name, Act\_Gender)
DIRECTOR (Dir\_id, Dir\_Name, Dir\_Phone)
MOVIES (Mov\_id, Mov\_Title, Mov\_Year, Mov\_Lang, Dir\_id)
MOVIE\_CAST (Act\_id, Mov\_id, Role)
RATING (Mov\_id, Rev\_Stars)
Write SQL queries to

- 1. List the titles of all movies directed by 'Hitchcock'.
- 2. Find the movie names where one or more actors acted in two or more movies.
- 3. List all actors who acted in a movie before 2000 and also in a movie after

2015 (use JOIN operation).

- 4. Find the title of movies and number of stars for each movie that has at least one rating and find the highest number of stars that movie received. Sort the result by movie title.
- 5. Update rating of all movies directed by 'Steven Spielberg' to 5.

CREATE DATABASE MOVIE; USE MOVIE;

CREATE TABLE ACTOR (

```
ACT ID INT,
ACT NAME VARCHAR (20),
ACT_GENDER CHAR (1),
PRIMARY KEY (ACT ID));
CREATE TABLE DIRECTOR (
DIR_ID INT,
DIR NAME VARCHAR (20),
DIR PHONE LONG,
PRIMARY KEY (DIR_ID));
CREATE TABLE MOVIES (
MOV ID INT,
MOV_TITLE VARCHAR (25),
MOV YEAR INT,
MOV_LANG VARCHAR (12),
DIR ID INT,
PRIMARY KEY (MOV_ID),
FOREIGN KEY (DIR_ID) REFERENCES DIRECTOR
(DIR_ID));
CREATE TABLE MOVIE CAST (
ACT ID INT,
MOV ID INT,
AROLE VARCHAR(10),
PRIMARY KEY (ACT ID, MOV ID),
FOREIGN KEY(ACT ID) REFERENCES ACTOR(ACT ID)
ON DELETE CASCADE,
```

```
FOREIGN KEY(MOV ID) REFERENCES
MOVIES(MOV ID) ON DELETE CASCADE);
CREATE TABLE RATING (
MOV ID INT,
REV STARS VARCHAR (25),
PRIMARY KEY (MOV_ID),
FOREIGN KEY (MOV_ID) REFERENCES MOVIES
(MOV_ID));
INSERT INTO ACTOR
VALUES (301, 'ANUSHKA', 'F'),
 (302, 'PRABHAS', 'M'),
 (303, 'PUNITH', 'M'),
 (304, 'JERMY', 'M');
select * from actor;
INSERT INTO DIRECTOR
 VALUES (60, 'RAJAMOULI', 8751611001),
  (61, 'HITCHCOCK', 7766138911),
  (62, 'FARAN', 9986776531),
  (63, 'STEVEN SPIELBERG', 8989776530);
select * from director;
INSERT INTO MOVIES
VALUES (1001, 'BAHUBALI-2', 2017, 'TELAGU', 60),
 (1002, 'BAHUBALI-1', 2015, 'TELAGU', 60),
 (1003, 'AKASH', 2008, 'KANNADA', 61),
```

```
(1004, 'WAR HORSE', 2011, 'ENGLISH', 63);
select * from movies;
INSERT INTO MOVIE CAST
VALUES (301, 1002, 'HEROINE'),
 (301, 1001, 'HEROINE'),
 (303, 1003, 'HERO'),
 (303, 1002, 'GUEST'),
 (304, 1004, 'HERO');
select * from movie_cast;
INSERT INTO RATING
VALUES (1001, 4),
 (1002, 2),
 (1003, 5),
 (1004, 4);
select * from rating;
----- List the titles of all movies directed by
'Hitchcock'.
SELECT MOV_TITLE
FROM MOVIES
WHERE DIR ID IN (SELECT DIR ID
FROM DIRECTOR
WHERE DIR NAME = 'HITCHCOCK');
```



----- Find the movie names where one or more actors acted in two or more movies.

SELECT MOV\_TITLE
FROM MOVIES M, MOVIE\_CAST MV
WHERE M.MOV\_ID=MV.MOV\_ID AND ACT\_ID IN
(SELECT ACT\_ID
FROM MOVIE\_CAST GROUP BY ACT\_ID
HAVING COUNT(ACT\_ID)>1)
GROUP BY MOV\_TITLE
HAVING COUNT(\*)>1;



--- List all actors who acted in a movie before 2000 and also in a movie after 2015 (use JOIN operation).

SELECT ACT\_NAME, MOV\_TITLE, MOV\_YEAR

FROM ACTOR A

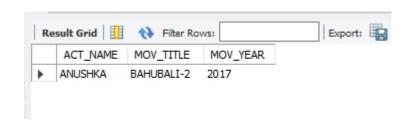
JOIN MOVIE\_CAST C

ON A.ACT\_ID=C.ACT\_ID

JOIN MOVIES M

ON C.MOV\_ID=M.MOV\_ID

WHERE M.MOV\_YEAR NOT BETWEEN 2000 AND 2015;



----- Find the title of movies and number of stars for each movie that has at least one rating and find the highest number of stars that movie received. Sort the result by movie title.

SELECT MOV\_TITLE, MAX(REV\_STARS)
FROM MOVIES
INNER JOIN RATING USING (MOV\_ID)
GROUP BY MOV\_TITLE
HAVING MAX(REV\_STARS)>0
ORDER BY MOV\_TITLE;



----- Update rating of all movies directed by 'Steven Spielberg' to 5 KL.

UPDATE RATING

SET REV\_STARS=5

WHERE MOV\_ID IN(SELECT MOV\_ID FROM MOVIES

WHERE DIR\_ID IN(SELECT DIR\_ID FROM DIRECTOR

WHERE DIR\_NAME = 'STEVEN SPIELBERG'));

## select \* from rating;

